CLAIMS

What is claimed is:

	-	1	
	i	A method	COMMISSING
T	1.	Amonou	comprising:

- 2 maintaining a first form of an intermediate result of an operation in a first register;
- maintaining a second form of the intermediate result in a second register;
- 4 responsive to receiving digits 1 to L-2 of the intermediate result from a digit
- 5 recurrence unit, where L represents a number of digits that satisfies a predetermined
- 6 precision for the operation, updating each of the first form and the second form of the
- 7 intermediate result by register swapping or concatenation under the control of load and
- 8 shift control logic and on-the-fly conversion logic;
- generating a rounded result by determining digits d_{L-1} and d_L and deriving from
- these the two digits d'_{L-1} and d'_{L} which are then appended to either the first form of the
- intermediate result or the second form of the intermediate result.
- 1 2. The method of claim 1, wherein the first form of the intermediate result comprises
- 2 a value R_j representing a sum of intermediate digits, $d_1 r^{-1} + d_2 r^{-2} + ... + d_j r^{-j}$, which
- 3 converges to an infinitely precise result as j tends to infinity, where r represents a radix
- 4 associated with a digit-recurrence algorithm implemented by the digit-recurrence unit, j
- 5 represents a current iteration of the digit recurrence algorithm, and d_i represents a digit
- 6 generated during iteration *j* of the digit-recurrence algorithm.
- 1 3. The method of claim 2, wherein the second form of the intermediate result
- comprises a value R_j^- , where R_j^- differs from R_j by a unit of the j^{th} position, r^{-j} .
- 1 4. The method of claim 3, wherein said updating each of the first form and the
- 2 second form of the intermediate result by register swapping or concatenation comprises
- 3 updating R_j and R_j^- based upon d_j , r, and R_{j-1} and R_{j-1}^- .

- 1 5. The method of claim 1, wherein storage typically allocated to a value, R_i^+ ,
- representing $R_j + r^{-j}$ in conventional digit-recurrence procedures employing on-the-fly 2
- rounding is saved by not relying upon and not maintaining R_i^+ during iterations 1 to L-2 3
- 4 of the digit-recurrence procedure.
- 1 6. The method of claim 1, wherein:
- $$\begin{split} R_{j} &= \begin{cases} (R_{j-1}, d_{j}) & d_{j} \geq 0 \\ (R_{j-1}^{-}, d_{j} + r) & d_{j} \leq -1 \end{cases}; \text{ and } \\ R_{j}^{-} &= \begin{cases} (R_{j-1}, d_{j} 1) & d_{j} \geq 1 \\ (R_{j-1}^{-}, d_{j} + r 1) & d_{j} \leq 0 \end{cases}. \end{split}$$
 2
- 3
- 1 7. The method of claim 1, wherein said generating a rounded result further
- 2 comprises:
- generating the rounded last digit d_L^{rnd} ; 3
- if $d_L^{rnd} \le -1$, then modifying d_L^{rnd} and d_{L-1} as follows: 4
- $d_{I}^{rnd} \longleftarrow r + d_{I}^{rnd}$, and 5
- $d_{L-1} \leftarrow -d_{L-1} -1$; and 6
- otherwise, if $d_{\scriptscriptstyle L}^{\it rnd} \geq r$, then modifying $d_{\scriptscriptstyle L}^{\it rnd}$ and $d_{\scriptscriptstyle L-1}$ as follows: 7
- $d_I^{rnd} \leftarrow -d_I^{rnd} r$, and 8
- $d_{I-1} \leftarrow -d_{I-1} + 1$. 9
- 1 8. The method of claim 1, wherein:
- 2 the digits are restricted to the digit set $\{-r+1, -r+2, ..., r-2\}$.
- 1 9. A method comprising:
- 2 receiving one or more operands upon which an operation is to be performed using
- 3 a digit-recurrence procedure employing on-the-fly rounding;
- providing a first storage location in which an intermediate result, R_i , of the 4
- operation is maintained as a sum of intermediate digits, $d_1 r^{-1} + d_2 r^{-2} + ... + d_j r^{-j}$, where 5

- 6 r represents a radix associated with the digit-recurrence procedure, j represents a current
- 7 iteration of the digit-recurrence procedure, and d_i represents a digit generated during
- 8 iteration *j* of the digit-recurrence procedure;
- 9 providing a second storage location in which a value, R_i^- , is maintained
- 10 representing $R_i r^{-j}$;
- during iterations 1 to L-2 of the digit-recurrence procedure, where L represents a
- number of digits that satisfies a predetermined precision for the operation, (1)
- generating d_j , and (2) updating R_j and R_j^- based upon d_j , r, and R_{j-1} and R_{j-1}^- ; and
- determining a rounded fractional result based upon (1) digits d_{L-1} and d_L and (2)
- 15 R_{L-2} or R_{L-2}^- .
- 1 10. The method of claim 9, wherein the operation comprises division.
- 1 11. The method of claim 9, wherein the operation comprises square-root.
- 1 12. The method of claim 9, wherein a value, R_i^+ , representing $R_i^- + r^{-j}$ is not
- 2 maintained during iterations 1 to L-2 of the digit-recurrence procedure, thereby saving
- 3 storage typically allocated to R_j^+ in conventional digit-recurrence procedures employing
- 4 on-the-fly rounding.
- 1 13. The method of claim 9, wherein the first and second storage locations comprise
- shift registers, and wherein said updating R_j and R_j^- based upon d_j , r, and R_{j-1} and
- 3 R_{i-1}^{-1} comprises shifting the contents of the shift registers and appending new digits.
- 1 14. The method of claim 9, wherein:
- 2 $R_{j} = \begin{cases} (R_{j-1}, d_{j}) & d_{j} \ge 0 \\ (R_{j-1}^{-}, d_{j} + r) & d_{j} \le -1 \end{cases}; \text{ and }$
- $R_{j}^{-} = \begin{cases} (R_{j-1}, d_{j} 1) & d_{j} \ge 1 \\ (R_{j-1}^{-}, d_{j} + r 1) & d_{j} \le 0 \end{cases}.$
- 1 15. The method of claim 9, wherein said determining a rounded fractional result

```
based upon (1) digits d_{L-1} and d_L and (2) R_{L-2} or R_{L-2}^- further comprises:

generating the digits d_{L-1} and d_L;

generating a rounded last digit d_L^{rnd};

if d_L^{rnd} \le -1, then modifying d_L^{rnd} and d_{L-1} as follows:

d_L^{rnd} \longleftarrow r + d_L^{rnd}, and

d_{L-1} \longleftarrow d_{L-1} -1; and

otherwise, if d_L^{rnd} \ge r, then modifying d_L^{rnd} and d_{L-1} as follows:
```

- 9 $d_L^{rnd} \longleftarrow d_L^{rnd} r, \text{ and}$
- $10 d_{L-1} \longleftarrow d_{L-1} + 1.$
- 1 16. The method of claim 9, wherein:
- the digits are constrained to the digit set $\{-r+1, -r+2, ..., r-2\}$.
- 1 17. A method comprising the steps of:
- 2 receiving one or more operands upon which an operation is to be performed using
- 3 a digit-recurrence procedure with on-the-fly rounding;
- 4 providing a first storage location in which an intermediate result of the
- 5 operation, R_i , is maintained, where,
- $R_{j} = d_{1} r^{-1} + d_{2} r^{-2} + \dots + d_{j} r^{-j}$ converges to an infinitely precise result as j
- 7 tends to infinity,
- 8 r represents a radix associated with the digit-recurrence procedure,
- *j* represents a current iteration of the digit-recurrence procedure, and
- d_j represents a digit generated during iteration j of the digit-recurrence
- 11 procedure;
- providing a second storage location in which a value, R_j^- , is maintained
- 13 representing $R_i r^{-j}$;
- an initialization step for constructing R_0 and R_0^- ;
- a main iteration step for generating digits d_1 through d_i and updating R_i and R_i^-

- during iterations 1 to L-2 of the iterative digit-recurrence procedure, where L represents a 16
- 17 number of digits that satisfies a predetermined precision for the operation; and
- a step for determining a rounded fractional result based upon (1) digits d_{L-1} and 18
- d_{L} and (2) R_{L-2} or R_{L-2}^{-} . 19
- The method of claim 17, wherein: 1

$$R_{j} = \begin{cases} (R_{j-1}, d_{j}) & d_{j} \ge 0 \\ (R_{j-1}^{-}, d_{j} + r) & d_{j} \le -1 \end{cases}; \text{ and } d_{j} \le 0$$

2
$$R_{j} = \begin{cases} (R_{j-1}, d_{j}) & d_{j} \ge 0 \\ (R_{j-1}^{-}, d_{j} + r) & d_{j} \le -1 \end{cases}; \text{ and}$$

$$R_{j}^{-} = \begin{cases} (R_{j-1}, d_{j} - 1) & d_{j} \ge 1 \\ (R_{j-1}^{-}, d_{j} + r - 1) & d_{j} \le 0 \end{cases}.$$

- 1 a 18. The method of claim 17, wherein said step for determining a rounded fractional
- 2 result based upon (1) digits d_{L-1} and d_L and (2) R_{L-2} or R_{L-2}^- further comprises:
- generating the digits d_{L-1} and d_L ; 3
- generating a rounded last digit d_I^{rnd} ; 4
- if $d_L^{rnd} \le -1$, then modifying d_L^{rnd} and d_{L-1} as follows: 5
- $d_I^{rnd} \longleftarrow r + d_I^{rnd}$, and 6
- $d_{r-1} \leftarrow -d_{r-1} -1$; and 7
- otherwise, if $d_{\scriptscriptstyle L}^{\it rnd} \geq r$, then modifying $d_{\scriptscriptstyle L}^{\it rnd}$ and $d_{\scriptscriptstyle L-1}$ as follows: 8
- $d_L^{rnd} \longleftarrow d_L^{rnd} r$, and 9
- $d_{L-1} \leftarrow -d_{L-1} + 1$. 10

1

1 18.

An apparatus comprising:

- 2 a first storage means for storing an intermediate result, R_i , of an operation on one
- 3 or more operands, the operation implemented as a digit recurrence procedure with on-the-
- 4 fly rounding, where $R_i = d_1 r^{-1} + d_2 r^{-2} + ... + d_j r^{-j}$, r represents a radix associated with
- 5 the digit-recurrence procedure, j represents a current iteration of the digit-recurrence
- 6 procedure, and d_i represents a digit generated during iteration j of the digit-recurrence
- 7 procedure;
- 8 a second storage means, coupled to the first storage means, for storing a
- 9 value, R_i^- , representing $R_i^- r^{-j}$;
- an update means, coupled to the first and second storage means, for updating
- 11 R_i and R_i^- based upon d_i , r, and R_{i-1} and R_{i-1}^- during iterations I to L-2 of the digit-
- 12 recurrence procedure;
- a digit selection means, coupled to the update means, for generating d_i during
- 14 iterations 1 to L-2 of the digit-recurrence procedure, where L represents a number of
- digits that satisfies a predetermined precision for the operation; and
- means, coupled to the first and second storage means, for determining a rounded
- fractional result based upon (1) digits d_{L-1} and d_L and (2) R_{L-2} or R_{L-2}^- .
- 1 $\sqrt[3]{20}$. The apparatus of claim 19, wherein the operation comprises a division operation
- 2 or a square-root operation.
- 1 $\sqrt[4]{21}$. The apparatus of claim 19, wherein the digit selection means supports both
- 2 division and square-root operations.
- The digit recurrence unit of claim 19, further comprising a delay element coupled
- 2 to the output of the digit selection means to hold at least two iterations of digits.

1



Įι

An apparatus comprising:

- 2 a first register to store an intermediate result, R_i , of an operation on one or more
- 3 operands, the operation implemented as a digit recurrence procedure with on-the-fly
- 4 rounding, where $R_j = d_1 r^{-1} + d_2 r^{-2} + ... + d_j r^{-j}$, r represents a radix associated with the
- 5 digit-recurrence procedure, *j* represents a current iteration of the digit-recurrence
- 6 procedure, and d_j represents a digit generated during iteration j of the digit-recurrence
- 7 procedure;
- 8 a second register, coupled to the first register, to store a value, R_i^- , representing
- $9 \qquad R_i r^{-j};$
- a digit selection lookup table to generating d_i during iterations 1 to L-2 of the
- digit-recurrence procedure, where L represents a number of digits that satisfies a
- 12 predetermined precision for the operation;
- load and shift control logic and on-the-fly conversion logic to update R_j and R_j^-
- based upon d_j , r, and R_{j-1} and R_{j-1}^- during iterations l to L-2 of the digit-recurrence
- 15 procedure; and
- final rounding logic to determine a rounded fractional result based upon (1) digits
- 17 d_{L-1} and d_L and (2) R_{L-2} or R_{L-2}^- .
- 1 $\sqrt[6]{24}$. The apparatus of claim 23, wherein the operation comprises a division operation
- 2 or a square-root operation.
- 1 25. The apparatus of claim 23, wherein the digit selection lookup table supports both
- 2 division and square-root operations.
- 1 $^{\sim}$ 26. The apparatus of claim 23, further comprising a delay element interposed between
- 2 the digit selection lookup table and the load and shift control logic and on-the-fly
- 3 conversion logic to hold at least two iterations of digits.

1.0

1 21. A machine-readable medium having stored thereon data representing sequences

2 of instructions, the sequences of instructions which, if executed by a processor, cause the

3 processor to:

4 maintain a first form of an intermediate result of an operation in a first register;

5 maintain a second form of the intermediate result in a second register;

6 update each of the first form and the second form of the intermediate result by

7 register swapping or concatenation responsive to receiving digits 1 to L-2 of the

8 intermediate result from a digit recurrence unit, where L represents a number of digits

9 that satisfies a predetermined precision for the operation; and

generate a rounded result by determining digits d_{L-1} and d_L and appending a

rounded last digit to either the first form of the intermediate result or the second form of

12 the intermediate result.

- 1 28. The machine-readable medium of claim 27, wherein the first form of the
- 2 intermediate result comprises a value R, representing a sum of intermediate digits,
- 3 $d_1 r^{-1} + d_2 r^{-2} + ... + d_j r^{-j}$, which converges to an infinitely precise result as j tends to
- 4 infinity, where r represents a radix associated with a digit-recurrence algorithm
- 5 implemented by the digit-recurrence unit, j represents a current iteration of the digit
- 6 recurrence algorithm, and d_j represents a digit generated during iteration j of the digit-
- 7 recurrence algorithm.
- 1² 29. The machine-readable medium of claim 28, wherein the second form of the
- 2 intermediate result comprises a value R_i^- , where R_i^- differs from R_i by a unit of the j^{th}
- 3 position, r^{-j} .